

An Overflow Problem in Network Coding for Secure Cloud Storage

Yu-Jia Chen, *Member, IEEE* and Li-Chun Wang, *Fellow, IEEE*,

National Chiao Tung University, Taiwan

Email: allan920693@g2.nctu.edu.tw and lichun@cc.nctu.edu.tw

Abstract

In this paper we define the overflow problem of a network coding storage system in which the encoding parameter and the storage parameter are mismatched. Through analyses and experiments, we first show the impacts of the overflow problem in a network coding scheme, which not only waste storage spaces, but also degrade coding efficiency. To avoid the overflow problem, we then develop the network coding based secure storage (NCSS) scheme. Thanks to considering both security and storage requirements in encoding procedures and distributed architectures, the NCSS can improve the performance of a cloud storage system from both the aspects of storage cost and coding processing time. We analyze the maximum allowable stored encoded data under the perfect secrecy criterion, and provide the design guidelines for the secure cloud storage system to enhance coding efficiency and achieve the minimal storage cost.

I. INTRODUCTION

Network coding is an attractive solution for secure cloud storage because of achieving the unconditional security. As long as protecting partial network coded data, a non-collusive eavesdropper cannot decode any symbol even with huge computing power for infinite time [1]. In principle, network coding simply mixes the data from different network nodes based on the well-designed linear combination rules. Hence, almost incurring no bandwidth expansion is another advantage of network coding.

A secure cloud storage using network coding is illustrated in Fig. 1. The original file is split into smaller chunks of symbols, each of which is encoded by Vandermonde matrix [2]. The different subsets of the encoded data are stored to two cloud databases. A legitimate user with access to two cloud databases can recover the entire original file. However, an eavesdropper with access to only one of the two cloud databases is unable to decode any of

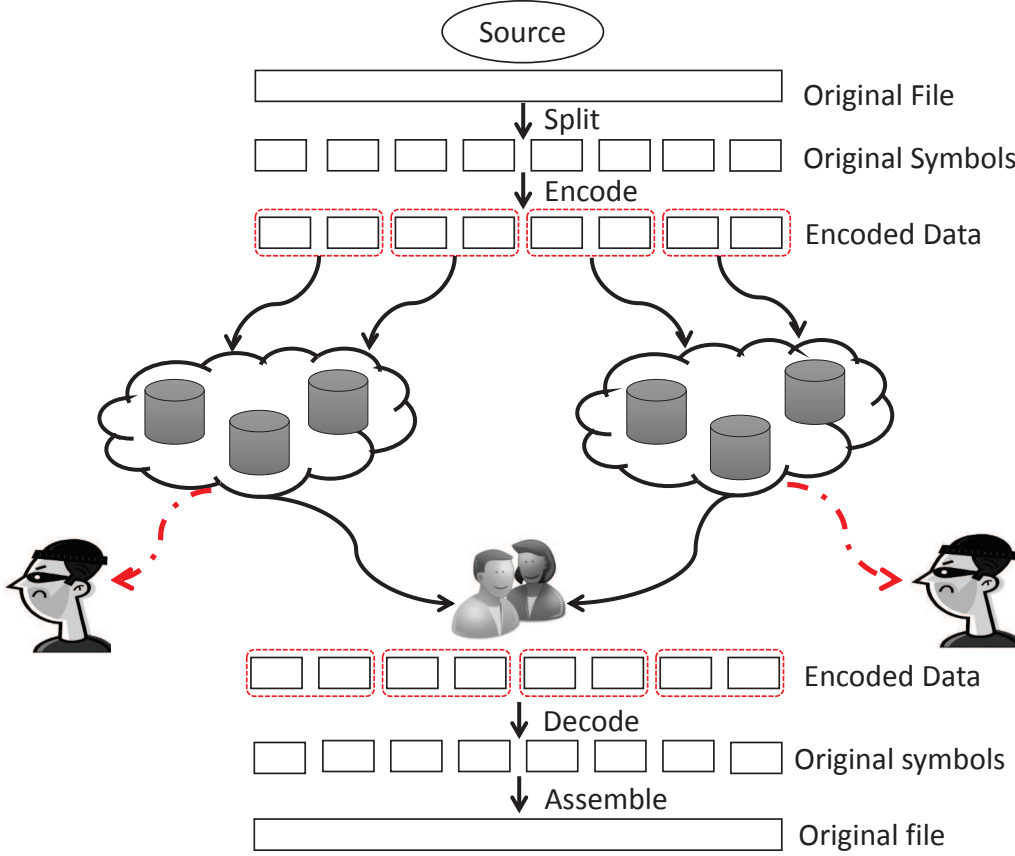


Fig. 1. An example of secure cloud storage using network coding.

the original symbols [3]. In summary, the network coding storage system consists of three procedures: splitting, encoding, and distributing to storage nodes.

Nevertheless, a secure network coding storage system may encounter a practical design issue when encoding parameters (such as the size of encoding matrix) is not jointly designed with the storage parameters (such as the storage size per node). If the mismatch between encoding and storage parameters occurs, it can cause bandwidth expansion and redundant computation cost. We coin the term the overflow problem for secure network coding storage system in this paper because the mismatch of encoding and storage parameters results in extended length of coded data in the format of digits.

Table I shows an example of the overflow problem for binary digits, where A is the encoding matrix for network coding, b_i and c_i are the original data, and network coded data, respectively. Assume that c_1 and c_3 are stored in the first database, and c_2 is stored in the second database. We can see that the bit length of coded c_i is larger than that of b_i for

TABLE I
EXAMPLE OF THE OVERFLOW PROBLEM FOR BINARY DIGITS

A	b	c
$\begin{bmatrix} 1 & 1 & 1 \\ 5 & 1 & 2 \\ 6 & 1 & 4 \end{bmatrix}$	$(0, 1, 1)^T$	$(2, 3, 5)^T = (10, 11, 101)^T$

$i = 1 \sim 3$. Also, the minimum bit length required for storing c_1 and c_3 is three in the first database, but in the second database the minimum bit length for storing c_2 is two.

To overcome the overflow problem, we propose a systematic design methodology to calculate the important system parameters of a network-coded cloud storage system. The major objective of the proposed scheme is to provide correct mapping between the encoding parameters and the storage parameters. The contributions of this work are explained as follows.

- Formulate the overflow problem of a network-coded cloud storage system, and perform experiments to show the impacts of the overflow problem, which not only waste storage spaces, but degrade computational efficiency.
- Propose a network coding based secure storage (NCSS) scheme to solve the overflow problem. To our best knowledge, the overflow problem for a network-coded storage system has not been investigated in the literature yet.
- Derive the upper bound of the amount of encoded data that can be stored in cloud databases to achieve the unconditional security level (i.e., perfect secrecy). Based on the derived upper bound, we present the analysis of storage cost minimization in the proposed NCSS scheme subject to different security levels.
- Finally, based on the experimental results, we suggest the design guidelines for determining important system parameters (e.g., the size of the encoding matrix) to accelerate the network coding process.

The rest of this paper is organized as follows. Section II describes related works. In Section III, we formulate the overflow problem in cloud storage using network coding. In Section IV,

we present the NCSS scheme. In Section V we give the security and storage analyses of the proposed scheme. Section VI shows the experimental results. Finally, we give our concluding remarks in Section VII.

II. BACKGROUND AND RELATED WORK

Network coding can be viewed as a generalized store-and-forward network routing principle. Messages from different source nodes are combined and regenerated at the intermediate nodes according to algebraic encoding. Besides the well-known advantages of throughput enhancement [4]–[6] and data robustness [7], the recent studies on network coding focus on reliability and security enhancement.

A. Network Coding for Data Recovery

Network coding can improve the efficiency of data recovery process when storage nodes fail in distributed storage systems. It was proved that the data recovery problem of distributed storage systems is equivalent to the multicasting problem of network routing [8]. The authors of [9] designed a cooperative network coding recovery mechanism for multiple node failures. A proxy-based multiple cloud storage system with the feature of fault tolerance was built based on the network coding storage scheme [10]. A network coding method called *Regenerating Code* was proposed to improve the repair process of distributed storage systems [11]. Different from erasure coding, the repaired data fragments are mixed in intermediate nodes, thereby reducing the repair bandwidth. The authors of [12] applied network coding to optimize the reliability performance of frequently accessed data in cloud storage systems.

B. Network Coding for Data Security

Another research area of network coding is to prevent data being eavesdropped during transmission. The information-theoretical security problem for an untrusted channel was first discussed in [13]. A network coding system was built to prove that a wiretapper cannot obtain any information from the transmitted message [14]. A weaker type of security issue was investigated in [15], where a node can decode packets only after receiving sufficient linear independent encoded data. The construction of a secure linear network code for a wiretap network was presented in [16].

The secrecy capacity for a network-coded cloud storage system was investigated in [17], [18], where the secrecy capacity is defined as the maximum amount of data that can be

securely stored under the perfect secrecy condition. The perfect secrecy condition ensure the eavesdropper cannot obtain any information of source data. The secrecy capacity for nodes with different storage capacities was derived in [19]. The coding scheme that can achieve the storage upper bound of secrecy capacity was proposed in [20]. The maximum data size being stored under the perfect secrecy condition for any number of eavesdropped nodes was determined in [21]. The authors of [22] considered how to achieve the information-theoretical secrecy when an eavesdropper can access some data in the storage nodes.

For secure storage over multiple clouds, similar to this work, the authors of [3] proposed a security protection scheme to ensure that no symbols can be decoded by an eavesdropper, which is weaker than perfect secrecy. In [23], a link eavesdropping problem in a network-coded cloud storage system was investigated. A publicly verifiable protocol for network coded cloud storage was proposed in [24].

C. Objective of This Paper

Different from the previous works focusing on the security and reliability enhancement of network coding, in this paper we focus on the storage efficiency and perfect security when applying network coding in multiple untrusted clouds. We define the overflow problem when network coding is applied in a cloud storage system, which has not been discussed previously. The overflow problem will result in extra extended encrypted data in the format of digits during encoding process, thereby increasing storage and computation cost. To overcome the overflow problem, we develop a systematic design methodology for calculating the encoding and storage parameters of a network-coded cloud storage system. Based on the proposed method, we further solve the storage cost optimization problem under the perfect secrecy constraint. The ultimate goal of this paper is to demonstrate that the performance of a network-coded cloud storage system can be improved by jointly designing the encoding and storage and parameters.

III. SYSTEM MODEL AND PROBLEM STATEMENT

Now we describe the coding scheme adopted in this paper and give the formal definition of the overflow problem.

A. Coding Scheme

Consider the original data vector $\mathbf{b} = (b_1, \dots, b_n)^T$ with base d , where elements b_i are independent random integers uniformly distributed over $\{0, \dots, d-1\}$. We use the terms

original data and plaintext data interchangeably in this paper. The goal of a cloud user is to securely store \mathbf{b} to multiple cloud databases. To achieve this goal, we adopt the same network coding scheme as [3], in which the input data are mapped to encoded symbols by linear transformation.

Denote \mathbf{A} as an $n \times n$ Vandermonde matrix, where $[A_{i,j}] = (a_j^{i-1})$. \mathbf{A} is used for the encoding matrix, where all the coefficients a_i are distinct nonzero elements over a finite field F_q , $q = 2^k > n$. Note that \mathbf{A} can be a $(n + m) \times n$ matrix, where the amount of redundancy m depends on the reliability requirement of the storage system.

A cloud user encodes data $\mathbf{c} = (c_1, \dots, c_n)^T = \mathbf{A}\mathbf{b}$ and splits the encoded data into p parts. We assume the cloud user can arbitrarily store any piece of the encoded data to any cloud database. Let $\tilde{\mathbf{c}}_i (i = 1, \dots, p)$ be the encoded data vector stored in the i -th cloud database. A legitimate user can collect $\tilde{\mathbf{c}}_i$ from the cloud databases and obtain the original data by performing $\mathbf{A}^{-1}\mathbf{c}$.

B. Security Model

Assume that an eavesdropper has infinite computing power, but can access only one cloud database. Also, it is assumed that the eavesdropper can have the full information about the encoding and decoding schemes, including the knowledge of the encoding matrix. The objective of an eavesdropper is to guess the original data. Although we consider only one eavesdropper in this paper, our result can be extended to the case of multiple eavesdroppers.

In our considered cloud storage system, every cloud database can support different security levels [25]. Denote P_{e_i} as the probability that the i -th cloud database can resist attacks. Also, the cloud user specifies a security requirement P_u , which represents the maximum probability that an eavesdropper can guess the original data. Next, we will show how to solve the overflow problem subject to the constraints of the security requirement when considering distributing encoded symbols to multiple cloud databases.

C. Overflow Problem

Although it was proved that the aforementioned network coding scheme can help prevent eavesdroppers from obtaining the information of the original data [1], the overflow problem occurs from the encoding process and the storage process. Specifically, if the encoding parameter and the storage parameter are mismatched, the length of encoded data in digital format may become larger than the length of the original data in digital format. As a result,

TABLE II
EXAMPLE OF THE DEFINITIONS FOR OVERFLOW PROBLEM

	A	b	c	$\tilde{c}_1 = (c_1, c_3)$	$\tilde{c}_2 = (c_2)$	strictly non-overflow	3-bounded non-overflow
Case1	$\begin{bmatrix} 1 & 1 & 1 \\ 5 & 1 & 2 \\ 6 & 1 & 4 \end{bmatrix}$	$(0, 1, 0)^T$	$(1, 1, 1)^T$	$(1, 1)$	(1)	Yes	Yes
Case2	$\begin{bmatrix} 1 & 1 & 1 \\ 5 & 1 & 2 \\ 6 & 1 & 4 \end{bmatrix}$	$(0, 1, 1)^T$	$(10, 11, 101)^T$	$(10, 101)$	(11)	No	Yes

storage spaces are wasted due to redundant encoded data. Now we formally state this problem by introducing the following definition.

Definition 1 (Strictly Non-overflow) Let $l_d(a)$ be the number of digits that represents a in base d . A piece of encoded data $\mathbf{c} = (c_1, \dots, c_n)^T$ is considered to be strictly non-overflow if and only if $l_d(c_i) \leq l_d(b_i)$ for every i . Thus, the length of the encoded data is equal to that of the plaintext data.

Definition 2 (α -bounded Non-overflow) Let $|\tilde{c}_i|$ denote the number of elements in \tilde{c}_i . A piece of encoded data $\mathbf{c} = (c_1, \dots, c_n)^T$ is considered to be α -bounded Non-overflow if and only if

$$\sum_{j=1}^{|\tilde{c}_i|} l_d(c_j) \leq |\tilde{c}_i| \alpha l_d(b_i) ,$$

for $1 \leq i \leq p$.

Assume the encoded data are randomly stored in cloud databases. Hence, the increasing cost of storage or computation resources caused by data extension can be measured by the extension degree $\alpha = \frac{l_d(c_i)}{l_d(b_i)}$ of the encoded data in the cloud database. Table II show the coding results for the two different overflow cases where $d = 2$ and $p = 2$. In case 1 there

TABLE III
NOTATIONS IN THIS PAPER

Notations	Descriptions
\mathbf{b}	Original data array
d	Base of b_i
$l_d(a)$	Number of digits that represents a in base d
\mathbf{A}	Encoding matrix
k	Use Galois Field size 2^k for \mathbf{A}
n	Matrix size of \mathbf{A}
p	Total number of cloud databases
\mathbf{c}	Encoded data vector
$\tilde{\mathbf{c}}_i$	Encoded data vector that stored in the i -th cloud database
$ \tilde{\mathbf{c}}_i $	Number of elements in $\tilde{\mathbf{c}}_i$
s_i	Number of digits in b_i
\mathbf{b}'	Regrouped data array
r	Size of \mathbf{b}'
P_{e_i}	Probability of the i -th cloud database can resist attacks
P_g	Probability that an eavesdropper can guess the original data
P_u	Security requirement: Maximum probability that an eavesdropper can guess the original data

are no redundant digits after the encoding process, but in case 2 the extension degree is bounded by 3.

IV. NETWORK CODING BASED SECURE STORAGE (NCSS) SCHEME

In this section, we analyze the overflow problem of a network-coded cloud storage system. We first give the criteria to choose the proper length of the data element to be encoded. Next, we present the data distribution method for achieving the required security level. Finally, we describe the system design methods of the NCSS scheme. Table III summarizes the notations used in this paper.

A. Coding Analysis

The following theorem can help calculate the encoding parameters to avoid the overflow problem of the secure network coding storage system.

Theorem 1 *Let s_i be the number of digits in b_i . Then, the system is strictly non-overflow if $s_i = s = \frac{k}{\log_2 d}$.*

Proof: First, we assume that $s_i < \frac{k}{\log_2 d}$. Then, we have

$$\frac{k}{\log_2 d} = k \log_d 2 = \log_d 2^k. \quad (1)$$

Because the coding process is manipulated with integers, we have $s_i \leq \log_d(2^k - 1)$. Since c_i is distributed over $\{0, \dots, 2^k - 1\}$, the maximum number of digits used to represent an encoded element is $l_d(c_i)_{\max} = \log_d(2^k - 1)$. Furthermore, the number of digits in b_i can be represented as $l_d(b_i)$. Thus, we have

$$s_i = l_d(b_i) \leq \log_d(2^k - 1) = l_d(c_i)_{\max}. \quad (2)$$

As a result, the overflow problem occurs because the length of encoded data may be larger than the length of the original data. Secondly, we assume that $s_i > \frac{k}{\log_2 d}$. We take exponentiation with base d on both sides and we have $d^{s_i} > d^{\log_d 2^k} = 2^k$ from (1). Since $b_i = d^{s_i}$, it contradicts the fact that the maximum value of b_i is $2^k - 1$. Hence, it follows that $s_i = s = \frac{k}{\log_2 d}$. ■

Theorem 2 *The system is α -bounded non-overflow if $s_i \geq \frac{1}{\alpha} \log_d(2^k - 1)$ for every i .*

Proof: Since $s_i = l_d(b_i)$ and $l_d(c_i)_{\max} = \log_d(2^k - 1)$, we have

$$\begin{aligned} \sum_{j=1}^{|\tilde{\mathbf{c}}_i|} l_d(c_j) &\leq |\tilde{\mathbf{c}}_i| \log_d(2^k - 1) \\ &= \alpha \cdot \frac{1}{\alpha} |\tilde{\mathbf{c}}_i| \log_d(2^k - 1) \\ &\leq \alpha |\tilde{\mathbf{c}}_i| s_i \\ &= \alpha |\tilde{\mathbf{c}}_i| l_d(b_i). \end{aligned} \quad (3)$$

Theorem 1 and 2 give the criteria of selecting the length of the plaintext data element. Next, we relate the security requirement to the amount of encoded stored data.

Theorem 3 *The system satisfies the security requirement P_u if*

$$\sum_{j=1}^{|\tilde{c}_i|} l_d(\tilde{c}_i(j)) \leq \sum_{t=1}^n l_d(c_t) + \log_d P_u - \log_d(1 - P_{e_i}) ,$$

for $1 \leq i \leq p$.

Proof: Recall that an eavesdropper can access only one cloud database. Hence, the probability P_g that an eavesdropper can guess the original data is the product of the invasion probability of the cloud database and the probability of guessing the remaining encoded digits. It follows that

$$\begin{aligned} P_g &= (1 - P_{e_i}) d^{-\left(\sum_{t=1}^n l_d(c_t) - \sum_{j=1}^{|\tilde{c}_i|} l_d(\tilde{c}_i(j))\right)} \\ &\leq (1 - P_{e_i}) d^{\log_d P_u - \log_d(1 - P_{e_i})} \\ &= P_u . \end{aligned} \tag{4}$$

■

B. System Design

The proposed NCSS scheme can be divided into three steps. First, a dynamic-length alphabet representation of network coded data is adopted based on Theorem 1 and Theorem 2. Second, the original data are preprocessed and regrouped before the encoding process. Third, the regrouped data are encoded and distributed to the corresponding cloud databases.

Figure 2 shows the system flow of the proposed NCSS scheme. Assume that a cloud user wants to store a single-digit data array $\mathbf{b} = (b_1, \dots, b_m)^T$ with base d to the p cloud databases. We first choose a power k for the field characteristics according to the following condition.

Condition 1 $2^k \geq d$

The field size must be larger than the maximal value of the data array element $d-1$. Otherwise, some data elements cannot be represented in the field. After that, a proper length of data elements s_i can be decided according to Theorem 1 and Theorem 2. This step is called dynamic length alphabet representation. We then regroup \mathbf{b} to $\mathbf{b}' = (b_1 \dots b_{s_1}, b_{s_1+1} \dots b_{s_1+s_2}, \dots, b_{\hat{s}_{r-1}+1} \dots b_{\hat{s}_r})$ based on the value of s_i , where $\hat{s}_r \triangleq \sum_{i=1}^r s_i$. Next, we generate an $n \times n$ encoding matrix \mathbf{A} with the following condition.

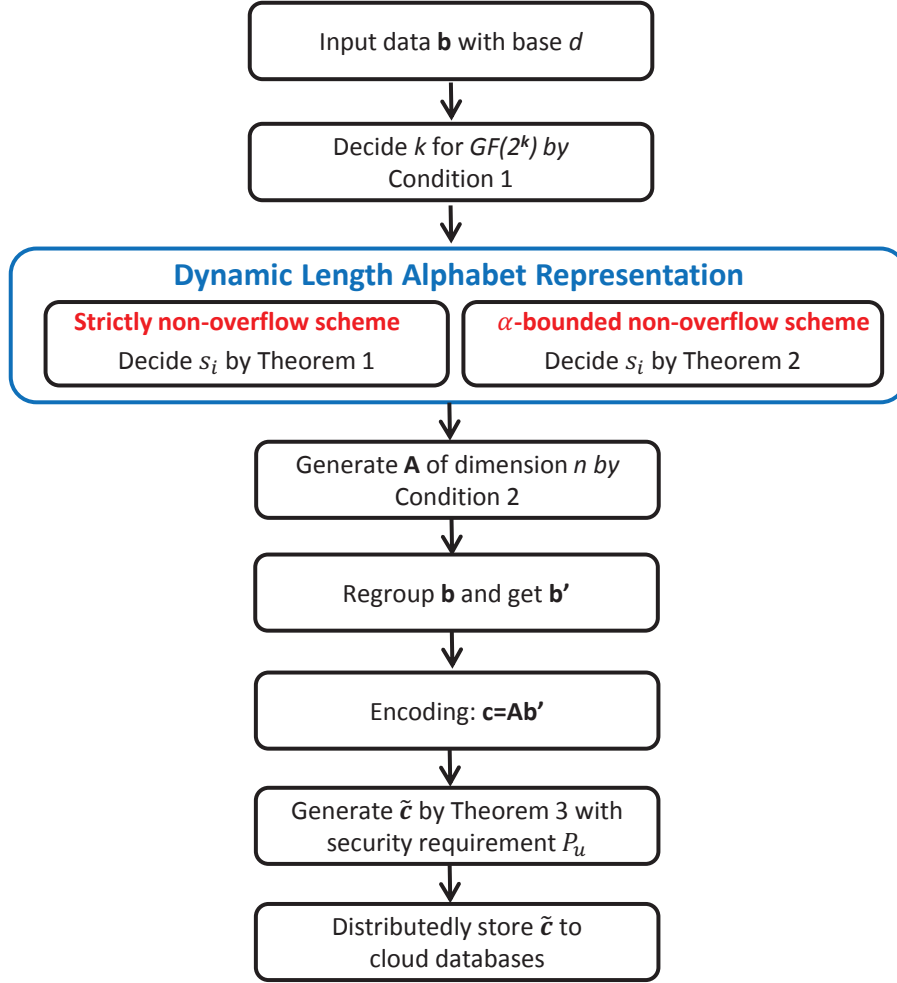


Fig. 2. System flow of NCSS scheme.

Condition 2 $n < 2^k$ and $n \leq r$

Since matrix \mathbf{A} is constructed from n distinct elements over the Galois Field, we have $n < 2^k$. In addition, the matrix multiplication cannot be operated if the size of encoding matrix is larger than the size of regrouped data array. We then encode \mathbf{b}' with \mathbf{A} and obtain the encoded data array $\mathbf{c} = (c_1, \dots, c_n)^T$. Finally, \mathbf{c} can be regrouped to $\tilde{\mathbf{c}}$ by Theorem 3, which specifies the maximum amount of encoded data that can be stored in a cloud database according to user's security requirement. Finally, the elements of $\tilde{\mathbf{c}}$ are distributed to the corresponding p cloud databases.

Table IV shows an example of the proposed NCSS scheme in the strictly non-overflow case. We assume that the original data is $\mathbf{b} = (0, 0, 1, 0, 1, 1, 1, 0, 1)$ and the encoded data are stored to two cloud databases with $P_{e_1} = 0.5$, $P_{e_2} = 0.25$, and $P_u = \frac{1}{64}$. From Theorem 3,

TABLE IV

EXAMPLE OF ADOPTING NCSS SCHEME IN STORING ENCODED DATA TO TWO CLOUD DATABASES

\mathbf{b}	d	k	s	\mathbf{b}'	r	n	\mathbf{A}	\mathbf{c}	$\tilde{\mathbf{c}}$
(0, 0, 1, 0, 1, 1, 1, 0, 1)	2	3	3	(001, 011, 101)	3	3	$\begin{bmatrix} 1 & 1 & 1 \\ 5 & 1 & 2 \\ 6 & 1 & 4 \end{bmatrix}$	(010, 100, 001)	(0101, 00001)

the maximal numbers of digits that can be stored in the first and the second cloud database are 4 and 5, respectively.

V. SECURITY ANALYSIS

In this section, we analyze the proposed NCSS scheme in terms of security level and storage cost. First, we discuss the issue of enhancing security level from a system design aspect. Then we derive the upper bound of data size that can be stored in the cloud under the constraint of perfect secrecy.

To begin with, from (4) we know that the lower bound of the security requirement P_u is

$$(1 - P_{e_i})d^{-\left(\sum_{t=1}^n l_d(c_t) - \sum_{j=1}^{|\tilde{\mathbf{C}}_i|} l_d(\tilde{c}_i(j))\right)} \leq P_u .$$

Since $l_d(c_t)$ is proportional to the size of Galois Field, it is anticipated that a larger size of encoding matrix n and a large value of power k of the field characteristics can result in a higher security level for the network coding storage system. However, increasing these encoding parameters can result extra coding complexity. Next, we show that the security level can be enhanced to the perfect secrecy by storing a certain amount of encoded data in the local machine. The notion of perfect secrecy represents that an eavesdropper can get no information of the original message.

Definition 3 (Perfect Secrecy Criterion [26]) *Let S denote the random variable associated with the secret data fragments and E denote the random variable associated encoded fragments observed by the eavesdropper. The perfect secrecy requires*

$$H(S|E) = H(S) ,$$

where $H(X)$ represents the entropy of a random variable X .

In the worst case, an eavesdropper can access the encoded data of all the cloud databases. The following theorem can be applied to specify the maximal amount of encoded data fragments that can be stored in the cloud, while keeping the rest of data in a local machine to ensure perfect secrecy.

Theorem 4 Assume that w -digit secret information is encoded with $n - w$ -digit data \mathbf{b} . For both strictly non-overflow and α -bounded non-overflow schemes, a cloud user can store at most $\sum_{j=1}^n l_d(c_j) - w$ digits of encoded data to the cloud under the perfect secrecy criterion.

Proof: Let $\mathbf{e}^{(h)}$ represent a subset containing any h components of vector \mathbf{e} . We use $\mathbf{e}_{i:j}$ to denote the subvector formed from the i -th to the j -th position of vector \mathbf{e} . The set of rows from the i -th to the j -th position of matrix \mathbf{D} is represented as $\mathbf{D}_{i:j}$. In addition, b_i are independent random variables uniformly distributed over \mathbb{F}_q with entropy $H(b_i) = H(b)$.

For simplicity, without loss of generality, assume that t contiguous components of the encoded data $\mathbf{c}_{p+1:p+t}$ are stored to the clouds. Then we can obtain

$$H(\mathbf{b}^{(w)}) = H(\mathbf{b}^{(w)} | \mathbf{c}_{p+1:p+t}) - H(\mathbf{b}^{(w)} | \mathbf{c}) \quad (5)$$

$$= I(\mathbf{b}^{(w)}; \mathbf{c}) - I(\mathbf{b}^{(w)}; \mathbf{c}_{p+1:p+t}) \quad (6)$$

$$= H(\mathbf{c}) - H(\mathbf{c}_{p+1:p+t}) - H(\mathbf{c} | \mathbf{b}^{(w)}) + H(\mathbf{c}_{p+1:p+t} | \mathbf{b}^{(w)}) \quad (7)$$

$$\leq H(\mathbf{c}) - H(\mathbf{c}_{p+1:p+t}) \quad (8)$$

In the above equations, (5) holds because of the perfect secrecy criterion and due to the fact that the secret information can be reconstructed if the entire codewords are given. In (8), we have $H(\mathbf{c}_{p+1:p+t} | \mathbf{b}^{(w)}) - H(\mathbf{c} | \mathbf{b}^{(w)}) \leq 0$ since

$$H(\mathbf{c} | \mathbf{b}^{(w)}) - H(\mathbf{c}_{p+1:p+t} | \mathbf{b}^{(w)}) = H(\mathbf{c}_{p+t+1:n} | \mathbf{b}^{(w)}, \mathbf{c}_{p+1:p+t}) \quad .$$

Since b_i are i.i.d random variables, it follows that

$$\begin{aligned} H(\mathbf{b}^{(w)}) &= H(b_{seq(1)}, b_{seq(2)}, \dots, b_{seq(w)}) \\ &= wH(b) \quad , \end{aligned} \quad (9)$$

where $seq(j)$ is the j -th element of a random integer sequence within the range 1 to n . Because the encoded data vector \mathbf{c} contains the entire information of \mathbf{b} at most, we can

obtain

$$H(\mathbf{c}) \leq nH(b) . \quad (10)$$

Moreover, the $n \times n$ Vandermonde matrix \mathbf{A} is nonsingular [2]. Thus the eavesdropper can apply the Gaussian elimination to obtain the reduced row echelon form of the submatrix \mathbf{S} , whose elements are $[S_{i,j}] = [A_{i,j}]$ for $p+1 \leq i, j \leq p+t$. The *Eavesdropper Reduced Matrix* \mathbf{M} can be obtained as

$$\mathbf{M}_{p+1:p+t} = \left[\begin{array}{cc|c|cc} m_1^p & \dots & & \dots & m_n^p \\ \vdots & \vdots & \mathbf{I}_t & \vdots & \vdots \\ m_1^{p+t-1} & \dots & & \dots & m_n^{p+t-1} \end{array} \right] , \quad (11)$$

where the other element of \mathbf{M} are the same as \mathbf{A} . Hence, the eavesdropper have t equations to solve n unknown elements. It implies that

$$H(\mathbf{c}_{p+1:p+t}) = tH(b) . \quad (12)$$

Substituting (9), (12) and (10) into (8), we obtain

$$tH(b) \leq nH(b) - wH(b) . \quad (13)$$

The above equation shows that we can store at most the $n - w$ components of encoded data to the clouds under perfect secrecy criterion. For the strictly non-overflow scheme, we have only one digit in each component of encoded data. Thus, we can store at most $\sum_{j=1}^n l_d(c_j) - w$ digits of encoded data to the clouds, while keeping the remaining w digits in the local machines. However, we may have multiple digits in each component of encoded data for α -bounded non-overflow scheme. Let $\mathbf{e}^{(\tilde{h})}$ represent a subset containing any w fragmentary components of vector \mathbf{e} . With at least n unknown digits, knowing $\mathbf{c}^{(\tilde{w})}$ cannot help solve \mathbf{b} . As a result, it follows that

$$I(\mathbf{c}^{(\tilde{w})}; \mathbf{b}) = 0 . \quad (14)$$

Not that we still have t equations to solve n unknown elements. That is,

$$H(\mathbf{b}^{(w)} | \mathbf{c}_{p+1:p+t}, \mathbf{c}^{(\tilde{w})}) = H(\mathbf{b}^{(w)} | \mathbf{c}_{p+1:p+t}) . \quad (15)$$

Finally, we obtain

$$I(\mathbf{c}_{p+1:p+t}, \mathbf{c}^{(\tilde{w})}; \mathbf{b}^{(w)}) = I(\mathbf{c}_{p+1:p+t}; \mathbf{b}^{(w)}) . \quad (16)$$

Consequently, we can select w digits of encoded data from different w components, i.e., select one digit for each component. These w -digit encoded data can be stored in the local machines, while the remaining $\sum_{j=1}^n l_d(c_j) - w$ digits are stored to the clouds. ■

VI. STORAGE MINIMIZATION

We are motivated to analyze the amount of stored encrypted data with the security requirement in terms of the probability that an eavesdropper can correctly guess the original data. This is because only a certain amount of encoded data fragments can be stored in the local machines to enhance the security level, as shown in the previous section. As the required security level increases, the amount of encoded data stored at the local site should increase.

A. Solving Storage Minimization Problem

Consider a cloud user keeps encoded data with length l in each encoding operation and stores the remaining encoded data to p cloud databases as shown in Fig. 3. We assume all the cloud databases have the same capability of preventing attacks (i.e., $P_{e_i}=P_e$) and the security requirement is P_u , which specifies the maximum probability that an eavesdropper can guess the original message. In addition to the encoded data, the encoding matrix is stored at the local site.

The storage cost at the local site is the function of encoding matrix size n and the amount of encoded data stored at a local machine for every encoding operation, denoted by l . Let m denote the length of the original message and α represent the number of encoding operations. Subject to a given security requirement P_u , the storage cost minimization problem is expressed as

$$\begin{aligned}
 & \text{minimize } f(n, l) = n^2 s + \alpha l \\
 & \text{subject to } (1 - P_e)^p d^{-\alpha l} \leq P_u \\
 & \quad 2 \leq n \leq 2^k \\
 & \quad l \leq n \\
 & \quad \alpha n s = m \\
 & \quad n, l \in \mathbb{Z}^+ ,
 \end{aligned} \tag{17}$$

where s is defined in Theorem 1. Note that an eavesdropper can guess the original message only if he/she can invade all the cloud databases and guess the encoded data in the local machine. It is observed that the optimization problem is nonconvex even if we relax the nonconvex constraints $n, l \in \mathbb{Z}^+$. The complete algorithm for solving this optimization problem is given in the Appendix.

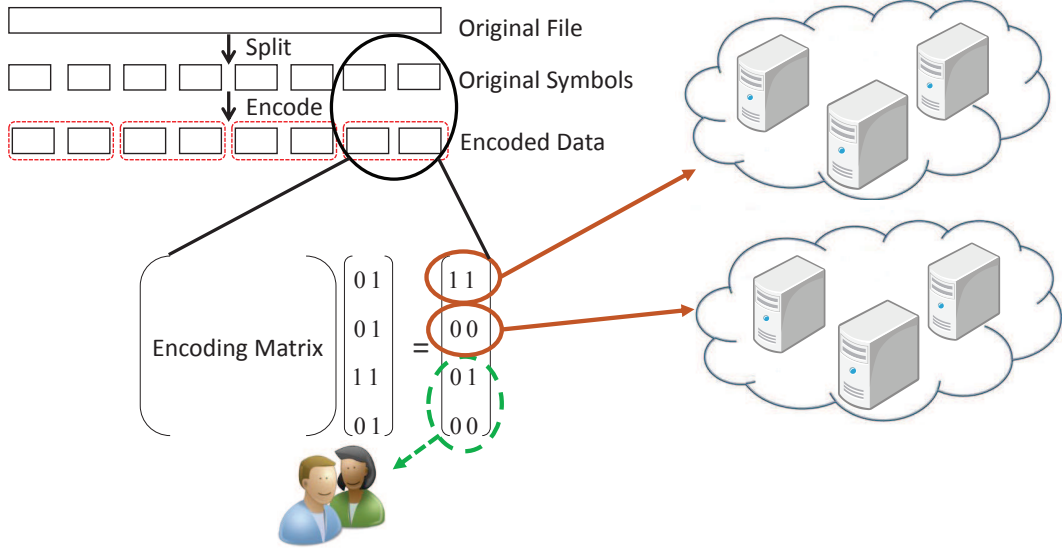


Fig. 3. Illustration of a user keeping a certain amount of encoded data at the local site to enhance security protection.

B. Discussions

Figure 4 shows the optimal parameter setting for encoding matrix size n versus the original message length m for $d = 2$, $P_e = 0.5$, $p = 3$, and $P_u = 10^{-6}$. As the message length increases, the size of the encoding matrix increases. A smaller encoding matrix size is preferred if Galois field size is large. Due to the integer constraints in the optimization problem, the encoding matrix size increases in a step-like function.

Figure 5 shows the storage cost $f(n, l)$ versus message length m for $d = 2$, $P_e = 0.5$, and $p = 3$. Intuitively, we need more storage space for lower P_u . However, the storage cost with various P_u are the same when m exceeds certain threshold. This is because the considered system is in the case of lower bound cost (i.e., $l = 1$). Noteworthy, a larger k can yield a smaller lower bound cost when $m > 1000$. In a general setting $k \in [8, 16]$ [27]. For $m < 1000$ it is suggested that the value of k is set to $k = 8$; otherwise, $k = 16$.

VII. EXPERIMENTAL RESULTS

Since the encoding process is performed on local machines, processing delay may be performance bottlenecks. Thus, it is of importance to investigate the impact of the system design parameters on the delay performance when considering a secure network coding storage scheme. We performed experiments on a commodity computer with an Intel Core i5 processor running at 2.4 GHz, 8 GB of RAM, and a 5400 RPM Hitachi 500 GB Serial ATA

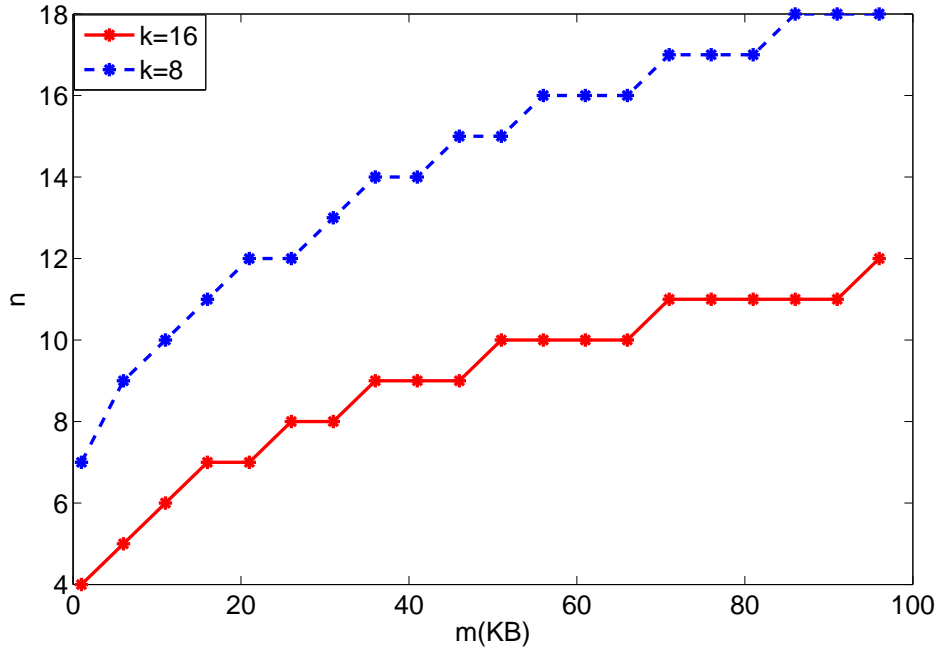


Fig. 4. Optimal parameter setting for encoding matrix size versus message length under different Galois Field sizes 2^k .

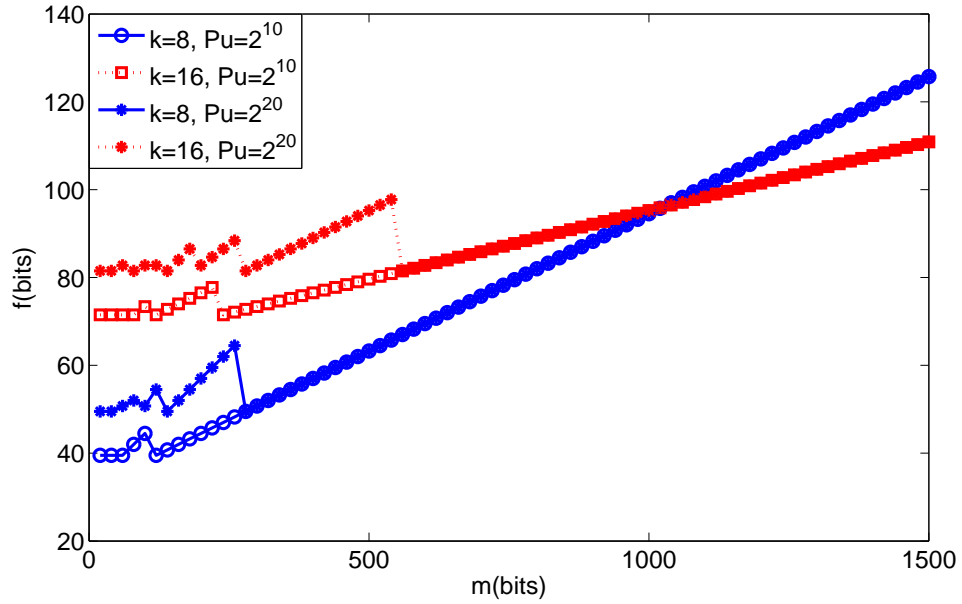


Fig. 5. Storage cost versus message length for different Galois Field sizes 2^k and security requirement P_u .

drive with an 8 MB buffer.

Figure 6 shows the multiplication processing time of the network coding storage system with different sizes of Galois Field. Although the complexity for the network coding is $O(n^2)$ modular multiplication, our result shows that the field size has only slight impact on the processing time, which supports our design methodology of selecting k . Specifically, it indicates the possibility that the security level can be enhanced significantly by selecting an appropriate design of k but only pay a very small computational cost.

Figure 7 shows the processing time between the strictly non-overflow and the α -bounded non-overflow schemes for 2MB file with $p = 2$, where $\alpha = 5$. The processing time is longer for a smaller n or k since the numbers of encoding times increase. As a result, the system spends more time in I/O operations and fetching data between the kernel and user [7]. Compared to the strictly non-overflow scheme, the α -bounded non-overflow scheme requires more computation cost. The α -bounded non-overflow scheme costs more than 11 times and 22 times of the processing time than that of the strictly non-overflow scheme when $k = 16$ and 8, respectively. Finally, the best performance is achieved when $n > 100$ for both non-overflow schemes. Because increasing n results in a larger cost than increasing k , we suggest to fix $n = 100$ and adjust k to meet the security requirements.

Figure 8 compares the processing time of the strictly non-overflow and the α -bounded non-overflow schemes versus the power of Galois Field characteristic k . As shown in the figure, the strictly non-overflow scheme is preferable to the α -bounded non-overflow scheme. It is noteworthy that k has negligible effect on the processing time of the strictly non-overflow scheme while it has a great impact on that of the α -bounded non-overflow scheme.

VIII. CONCLUSIONS

In this paper, we investigated the overflow problem in a network coding cloud storage system. When the overflow problem occurs, it does not only require more storage spaces but increases the processing time in encoding. We developed the network coding based secure storage (NCSS) scheme. A systematic approach for the optimal encoding and storage parameters was provided to solve the overflow problem and minimize the storage cost. We also derived an analytical upper bound for the maximal allowable stored data in the cloud nodes under perfect secrecy criterion. Our experimental results demonstrated that encoding efficiency in terms of processing time can be improved by jointly design of the encoding and the storage system parameters. More importantly, we suggested the key design guidelines for

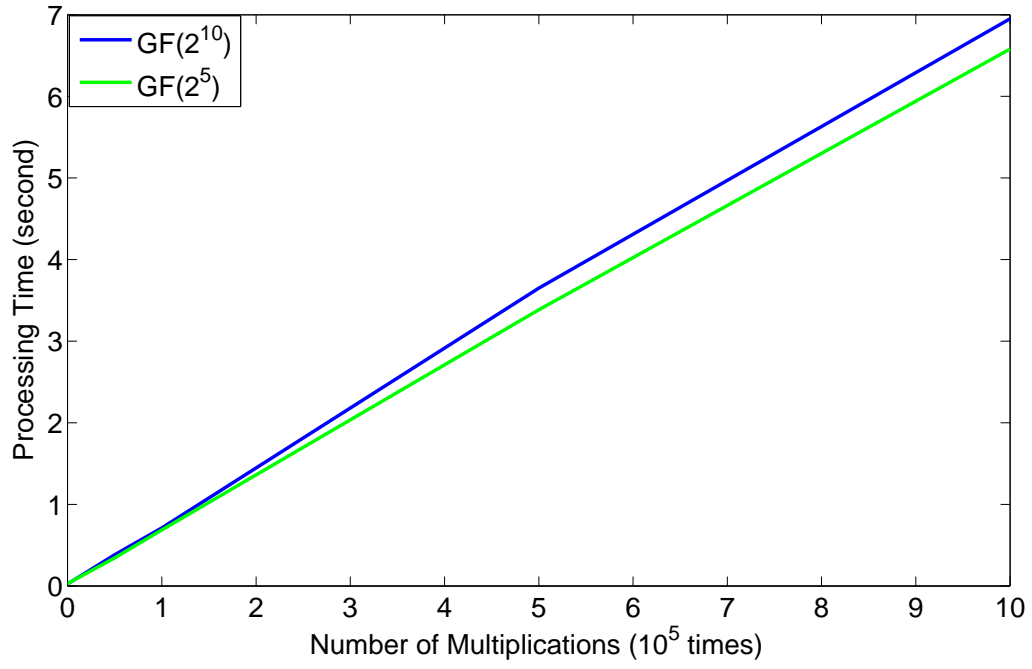


Fig. 6. Processing time versus the multiplication times for different Galois Fields 2^k .

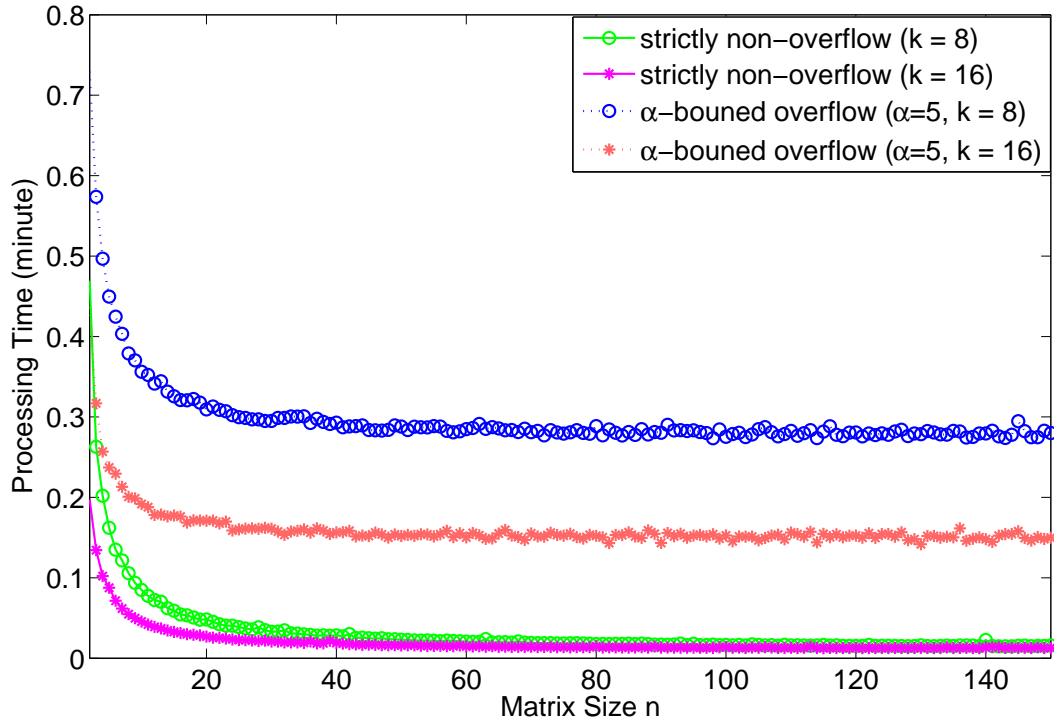


Fig. 7. Comparison of processing time between the strictly non-overflow and the α -bounded non-overflow schemes versus matrix size n with $p = 2$.

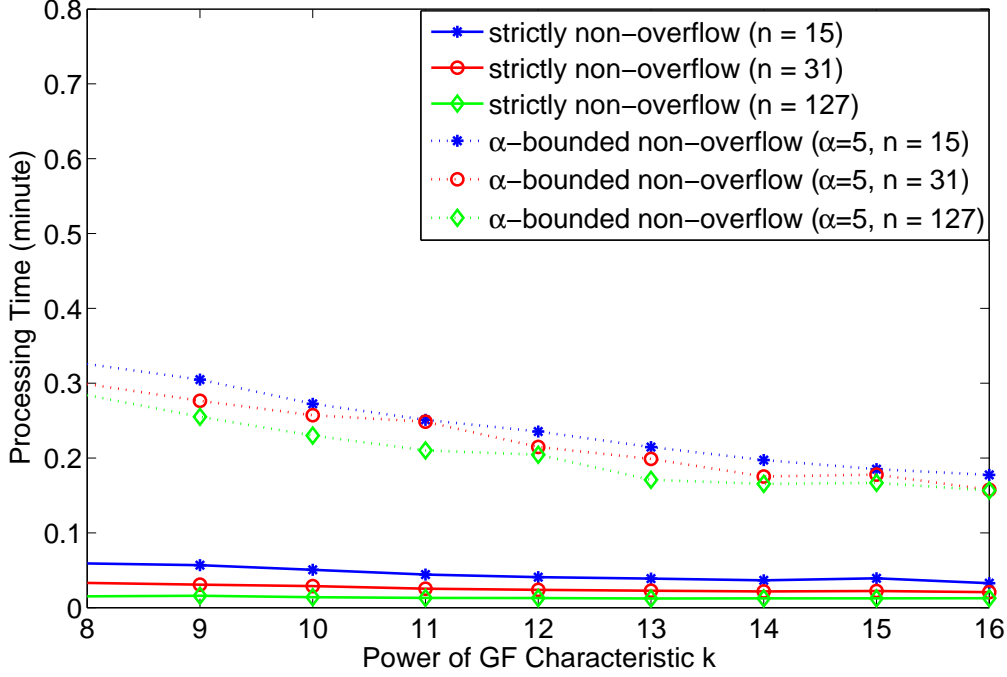


Fig. 8. Comparison of processing time between the strictly non-overflow and the α -bounded non-overflow schemes versus power of Galois Field characteristic k with $p = 2$.

secure network coding storage systems to optimize the performance tradeoff among security requirement, storage cost per node, and encoding processing time. In the future research, we will further incorporate the factors of user budgets and file recovery into the secure network coding distributed storage system.

REFERENCES

- [1] P. F. Oliveira, L. Lima, T. T. V. Vinhoza, J. Barros, and M. Medard, "Trusted storage over untrusted networks," *IEEE Global Communication Conference*, 2010.
- [2] A. Klinger, "The Vandermonde matrix," *The American Mathematical Monthly*, 1967.
- [3] P. F. Oliveira, L. Lima, T. T. Vinhoza, J. Barros, and M. Medard, "Coding for trusted storage in untrusted networks," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1890–1899, 2012.
- [4] W. Qiao, J. Li, and J. Ren, "An efficient error-detection and error-correction scheme for network coding," *IEEE Global Telecommunications Conference*, pp. 1–5, 2011.
- [5] D. Zeng, S. Guo, Y. Xiang, and H. Jin, "On the throughput of two-way relay networks using network coding," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 1, pp. 191–199, 2014.
- [6] Y. Wu and S.-Y. Kung, "Distributed utility maximization for network coding based multicasting: A shortest path approach," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1475–1488, 2006.
- [7] C. Fragouli and J. L. Boudec, "Network coding: An instant primer," *ACM SIGCOMM Computer*, vol. 36, no. 1, pp. 63–68, 2006.

- [8] A. Dimakis, P. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4539–4551, 2010.
- [9] Y. Hu, Y. Xu, X. Wang, C. Zhan, and P. Li, "Cooperative recovery of distributed storage systems from multiple losses with network coding," *IEEE Journal on Selected Areas in Communications*, vol. 28, pp. 268–276, 2010.
- [10] Y. Hu, H. Chen, P. Lee, and Y. Tang, "NCcloud: Applying network coding for the storage repair in a cloud-of-clouds," in *Proc. of the 10th USENIX Conf. on File and Storage Tech*, vol. 1, 2012.
- [11] D. S. Papailiopoulos, J. Luo, A. G. Dimakis, C. Huang, and J. Li, "Simple regenerating codes: Network coding for cloud storage," *IEEE International Conference on Computer Communications*, pp. 2801–2805, 2012.
- [12] Y. Lu, J. Hao, X.-J. Liu, and S.-T. Xia, "Network coding for data-retrieving in cloud storage systems," pp. 51–55, 2015.
- [13] L. Ozarow and A. Wyner, "Wire-tap channel II," *Advances in Cryptology*, pp. 33–50, 1985.
- [14] N. Cai and R. Yeung, "Secure network coding," *IEEE International Symposium on Information Theory*, p. 323, 2002.
- [15] K. Bhattad and K. Narayanan, "Weakly secure network coding," *Workshop on Network Coding, Theory, and Application*, 2005.
- [16] N. Cai and R. W. Yeung, "Secure network coding on a wiretap network," *IEEE Transactions on Information Theory*, vol. 57, no. 1, pp. 424–435, 2011.
- [17] S. Pawar, S. El Rouayheb, and K. Ramchandran, "On secure distributed data storage under repair dynamics," *IEEE International Symposium on Information Theory Proceedings*, pp. 2543–2547, 2010.
- [18] N. B. Shah, K. Rashmi, and P. V. Kumar, "Information-theoretically secure regenerating codes for distributed storage," *IEEE Global Telecommunications Conference*, pp. 1–5, 2011.
- [19] T. Ernvall, S. El Rouayheb, C. Hollanti, and H. V. Poor, "Capacity and security of heterogeneous distributed storage systems," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 12, pp. 2701–2709, 2013.
- [20] A. S. Rawat, N. Silberstein, O. O. Koyluoglu, and S. Vishwanath, "Secure distributed storage systems: Local repair with minimum bandwidth regeneration," *International Symposium on Communications, Control and Signal Processing*, pp. 5–8, 2014.
- [21] S. Goparaju, S. E. Rouayheb, R. Calderbank, and H. V. Poor, "Data secrecy in distributed storage systems under exact repair," *International Symposium on Network Coding*, pp. 1–6, 2013.
- [22] N. Shah, K. Rashmi, and P. Kumar, "Information-theoretically secure regenerating codes for distributed storage," *IEEE Global Communication Conference*, 2011.
- [23] Y.-J. Chen, L.-C. Wang, and C.-H. Liao, "Eavesdropping prevention for network coding encrypted cloud storage systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, pp. 2261–2273, 2016.
- [24] F. Chen, T. Xiang, Y. Yang, and S. S. Chow, "Secure cloud storage meets with secure network coding," *IEEE Transactions on Computers*, vol. 65, no. 6, pp. 1936–1948, 2016.
- [25] M. Barua, X. Liang, R. Lu, and X. Shen, "ESPAC: Enabling security and patient-centric access control for eHealth in cloud computing," *International Journal of Security and Networks*, vol. 6, no. 2, pp. 67–76, 2011.
- [26] J. L. Massey, "An introduction to contemporary cryptology," *Proceedings of the IEEE*, vol. 76, no. 5, pp. 533–549, 1988.
- [27] G. Angelopoulos, M. Médard, and A. P. Chandrakasan, "Energy-aware hardware implementation of network coding," *International Conference on Research in Networking*, pp. 137–144, 2011.

APPENDIX

Here we first show that the original storage cost minimization (17) is not convex even when the integer constraint is relaxed. Then we give the algorithm for solving the optimization problem by minimizing over separated variables.

Theorem 5 *The objective function of the original storage cost minimization (17) is not convex.*

Proof: We consider the case of strictly non-overflow scheme. Substituting $s_i = s = \frac{k}{\log_2 d}$ into (17), the original storage cost minimization is equivalent to

$$\begin{aligned} & \text{minimize } \tilde{f}(n, l) = \frac{k}{\log_2 d} n^2 + \frac{m \log_2 d}{k} n^{-1} l \\ & \text{subject to } -\frac{k}{m \log_2 d} \log_d \frac{P_u}{(1 - P_c)^p} n \leq l \leq n \\ & \quad 2 \leq n \leq 2^k \\ & \quad n, l \in \mathbb{Z}^+ . \end{aligned} \tag{18}$$

Then we prove the theorem by showing that the Hessian matrix of the objective function is not positive semidefinite. The Hessian matrix of $\tilde{f}(n, l)$ is

$$H(\tilde{f}) = \begin{bmatrix} 2a + 2bln^{-3} & -bn^{-2} \\ -bn^{-2} & 0 \end{bmatrix} ,$$

where $a = \frac{k}{\log_2 d} > 0$ and $b = \frac{m \log_2 d}{k} > 0$. Then, we solve the characteristic equation

$$\det(H(\tilde{f}) - \lambda I) = \lambda^2 - (2a + 2bln^{-3})\lambda - b^2 n^{-4} = 0 .$$

We can obtain

$$\lambda = \frac{2a + 2bln^{-3} \pm \sqrt{(2a + 2bln^{-3})^2 + 4b^2 n^{-4}}}{2} .$$

Since the eigenvalues of $H(\tilde{f})$ is not all positive, $H(\tilde{f})$ is not positive semidefinite. Thus \tilde{f} is not convex. ■

We are now ready for solving the equivalent optimization problem (18) by minimizing over separated variables. Define $\tilde{f}^*(n, l) \triangleq \min_{n \in \mathbf{B}, l \in \mathbf{A}} \tilde{f}$ and $l^* \triangleq \arg \min_{l \in \mathbf{A}} \tilde{f}(n, l)$, where $\mathbf{A} = \{x | x \in \mathbb{Z}^+, \frac{nk}{m \log_2 d} \log_d \frac{P_u}{(1 - P_c)^p} \leq x < n\}$ and $\mathbf{B} = \{x | x \in \mathbb{Z}^+, 2 \leq x < 2^k\}$. We first minimize over n

$$\tilde{f}^*(n, l) = \min_{n \in \mathbf{B}} \{x | x = \min_{l \in \mathbf{A}} \tilde{f}(n, l)\} .$$

Since $\min_{l \in A} \tilde{f}(n, l)$ is a linear function with one variable in \mathbb{Z}_{++}^1 for fixed n and the coefficient is positive, we obtain

$$l^* = \min\{\mathbf{A}\} = \left\lceil \frac{-nk}{m \log_2 d} \log_d \frac{P_u}{(1 - P_c)^p} \right\rceil .$$

As a result, we can solve the optimization problem iteratively as:

- Step 0: Initiate $\mathbf{C} = \emptyset$ and $\mathbf{B} = \{x | x \in \mathbb{Z}^+, 2 \leq x < 2^k\}$.
- Step 1: Select $n \in \mathbf{B}$ and set $l = \left\lceil \frac{-nk}{m \log_2 d} \log_d \frac{P_u}{(1 - P_c)^p} \right\rceil$.
- Step 2: Calculate $c = \tilde{f}(n, l)$.
- Step 3: Set $\mathbf{C} = \mathbf{C} \cup \{c\}$ and $\mathbf{B} = \mathbf{B} - \{n\}$.
- Step 4: Iterate 1 to 4 until $\mathbf{B} = \emptyset$.
- Step 5: Obtain $\tilde{f}^*(n, l) = \min\{\mathbf{C}\}$.



Yu-Jia Chen received the B.S. degree and Ph.D. degree in electrical engineering from National Chiao Tung University, Taiwan, in 2010 and 2016, respectively. He is currently a postdoctoral fellow in National Chiao Tung University. His research interests include network coding for secure storage in cloud datacenters, software defined networks (SDN), and sensors-assisted applications for mobile cloud computing. Yu-Jia Chen has published 15 conference papers and 3 journal papers. He is holding two US patent and three ROC patent.



Li-Chun Wang (M'96 – SM'06 – F'11) received the B.S. degree from National Chiao Tung University, Taiwan, R.O.C. in 1986, the M.S. degree from National Taiwan University in 1988, and the Ms. Sci. and Ph. D. degrees from the Georgia Institute of Technology, Atlanta, in 1995, and 1996, respectively, all in electrical engineering.

From 1990 to 1992, he was with the Telecommunications Laboratories of Chunghwa Telecom Co.

In 1995, he was affiliated with Bell Northern Research of Northern Telecom, Inc., Richardson, TX.

From 1996 to 2000, he was with AT&T Laboratories, where he was a Senior Technical Staff Member in the Wireless Communications Research Department. Since August 2000, he has joined the Department of Electrical and Computer Engineering of National Chiao Tung University in Taiwan and is the current Chairman of the same department. His current research interests are in the areas of radio resource management and cross-layer optimization techniques for wireless systems, heterogeneous wireless network design, and cloud computing for mobile applications.

Dr. Wang won the Distinguished Research Award of National Science Council, Taiwan in 2012, and was elected to the IEEE Fellow grade in 2011 for his contributions to cellular architectures and radio resource management in wireless networks. He was a co-recipient (with Gordon L. Stuber and Chin-Tau Lea) of the 1997 IEEE Jack Neubauer Best Paper Award for his paper "Architecture Design, Frequency Planning, and Performance Analysis for a Microcell/Macrocell Overlaying System," IEEE Transactions on Vehicular Technology, vol. 46, no. 4, pp. 836-848, 1997. He has published over 200 journal and international conference papers. He served as an Associate Editor for the IEEE Trans. on Wireless Communications from 2001 to 2005, the Guest Editor of Special Issue on "Mobile Computing and Networking" for IEEE Journal on Selected Areas in Communications in 2005, "Radio Resource Management and Protocol Engineering in Future Broadband Networks" for IEEE Wireless Communications Magazine in 2006, and "Networking Challenges in Cloud Computing Systems and Applications," for IEEE Journal on Selected Areas in Communications in 2013, respectively. He is holding 10 US patents.